

Claims

What is claimed is:

1. A method performed by a computer system comprising:
detecting a test module interface associated with a test module; and
calling a function identified by the test module interface to cause a first test
configuration of the test module to be created.
2. The method of claim 1, further comprising:
detecting a test routine associated with the test module using the first test
configuration; and
causing the test routine to be executed.
3. The method of claim 1, further comprising:
loading the test module;
detecting a device associated with the test module; and
in response to detecting the device, creating the test module interface.
4. The method of claim 3, further comprising:
detecting a change associated with the device; and
in response to detecting the change, reinstalling the test module interface associated
with the test module.

PATENT

Docket No.: DC-02461 (16356.549)

- 1 5. The method of claim 4, further comprising:
2 in response to detecting the change,
3 calling the function to cause a second test configuration of the test module to be
4 created.
- 1 6. The method of claim 1, further comprising:
2 registering a use of the test module by a program.
- 1 7. The method of claim 6, further comprising:
2 unloading the test module; and
3 informing the program of the unloading prior to unloading the test module.
- 1 8. The method of claim 7, further comprising:
2 conveying a defer signal from the program to the test module; and
3 in response to the defer signal, canceling the unloading of the test module.
- 1 9. A computer system comprising:
2 a processor; and
3 a memory coupled to the processor and including a program and a test module;
4 the program being executable by the processor to:
5 detect a test module interface associated with a test module; and
6 call a function identified by the test module interface to cause a first test
7 configuration of the test module to be created.
- 1 10. The computer system of claim 9, wherein the program is executable to:
2 detect a test routine associated with the test module using the first test configuration;
3 and
4 cause the test routine to be executed.

PATENT

Docket No.: DC-02461 (16356.549)

- 1 11. The computer system of claim 9, wherein the test module is executable to:
2 detect a device associated with the test module; and
3 in response to detecting the device, create the test module interface.
- 1 12. The computer system of claim 11, wherein the test module is executable to:
2 detect a change associated with the device; and
3 in response to detecting the change, reinstall the test module interface associated with
4 the test module.
- 1 13. The computer system of claim 12, wherein the program is executable to:
2 call the function identified by the test module interface to cause a second test
3 configuration of the test module to be created.
- 1 14. The computer system of claim 9, wherein the program is executable to:
2 cause a use of the test module by the program to be registered.
- 1 15. The computer system of claim 14, wherein the test module is executable to:
2 cause the test module to be unloaded; and
3 cause the program to be notified prior to unloading the test module.
- 1 16. The computer system of claim 15, wherein the program is executable to:
2 convey a defer signal from to the test module; and
3 wherein the test module is executable to:
4 in response to the defer signal, cause the unloading of the test module to be
5 canceled.

PATENT

Docket No.: DC-02461 (16356.549)

1 17. A computer system comprising:

2 a processor; and

3 a memory coupled to the processor and including a first program and a second
4 program that includes an interface;

5 the first program being executable by the processor to:

6 call a first function identified by the interface, the first function being

7 executable by the processor to register a use of the second program by
8 the first program.

1 18. The computer system of claim 17, wherein the first function is executable by the
2 processor to cause an entry associated with the first program to be stored in the memory.

1 19. The computer system of claim 18, wherein the entry includes a first identifier
2 associated with the first program, a second identifier associated with the interface, and a third
3 identifier associated with a second function.

1 20. The computer system of claim 17, wherein the second program is executable by the
2 processor to:

3 cause the second program to be unloaded from the memory; and

4 cause the first program to be notified that the second program seeks to be unloaded.

1 21. The computer system of claim 20, wherein the first program is executable by the
2 processor to:

3 convey a defer signal to the second program in response to being notified that the

4 second program seeks to be unloaded; and

5 wherein the second program is executable by the processor to:

6 cause the second program not to be unloaded in response to receiving the defer signal.

